# Introduction

## Généralité

|  |  |  |
| --- | --- | --- |
| **Projet** | | |
| **Nom du projet** | EasyGoing | |
| **Début du projet** | 14.09.2015 | |
| **Fin du projet** | 04.01.2016 | |
| **Durée effective** | 90 heures | |
| **Acteurs** | | |
| **Répondant** | Dr. René Rentsch | rene.rentsch@heig-vd.ch |
| **Développeurs** | Miguel Santamaria (Chef de projet)  Raphaël Racine (Suppléant)  Thibaud Duchoud  Michelle Vanessa Meguep Sakam  Karim Ghozlani | [miguel.santamaria@heig-vd.ch](mailto:miguel.santamaria@heig-vd.ch)  [raphael.racine@heig-vd.ch](mailto:raphael.racine@heig-vd.ch)  [thibaud.duchoud@heig-vd.ch](mailto:thibaud.duchoud@heig-vd.ch)  [michelle.meguepsakam@heig-vd.ch](mailto:michelle.meguepsakam@heig-vd.ch)  karim.ghozlani@heig-vd.ch |

# Objectifs du projet

Ce projet a pour objectif le développement d’un site web permettant la gestion de projets.

Il permet aux membres d’une équipe de gérer un projet autour d’un « scrum board ». Il offre la possibilité de visualiser différentes tâches, qui travaille sur l’une de ces dites tâches et l’avancement de celle-ci.

Des événements ont été mis en place et ceux-ci permettent de suivre les différentes actions effectuées par un utilisateur (par exemple lors du déplacement d’une tâche ou de l’ajout d’un membre au projet). Ce système est conçu comme un fil d’actualité.

Notre projet dispose également d’un tutoriel d’utilisation qui facilite grandement la tâche des nouveaux venus et d’une grande ergonomie

# Technologies utilisées et conventions de codage

### Technologies

Voici la liste des technologies utilisées :

* **Apache v2.4** comme serveur http.
* **PHP v5.6** comme langage coté serveur.
  + **Zend Framework 2** comme framework php.
* **MySQL v5.6** comme moteur de base de données.
  + **PHPMyAdmin v4.4 et MySQL Workbench v6.3** pour gérer la base de données.
* **HTML – CSS** comme langages de structure et design des pages web.
* **Javascript/JQuery v2.1** comme langage/bibliothèque pour dynamiser les pages web.
* **Différentes bibliothèques Javascript** (comme Bootstrap) pour réaliser des actions spécifiques concernant la gestion des boards principalement (cliquer-déposer, animation, gestion d’une grille …).

### Conventions de codage

##### Général

Les conventions de codage que nous avons utilisées sont inspirées de celles utilisées dans le framework Zend et qui sont disponibles sur internet (<http://framework.zend.com/manual/1.12/fr/coding-standard.naming-conventions.html>).

Tout doit être écrit en anglais.

#### Nommage

Les noms de fonctions, de variables et de classes doivent être descriptifs : il faut éviter les abréviations.

Les noms des fichiers PHP correspondent à la classe qu’il contient.

Les variables et fonctions sont écrites en camelCase (première lettre en minuscule et chaque nouveau mot commence par une majuscule).

Pour les variables de classes (membres), si elles sont « protected » ou « private » ont les précède d’un « \_ ».

Les méthodes respectent les mêmes conventions que les variables de classes.

Les constantes sont écrites en majuscule et chaque mot est séparé par un « \_ ».

Les classes sont écrites en PascalCase (chaque mot commence par une majuscule).

#### Autres

En PHP, la première accolade suivant une instruction se place après un retour à la ligne. En Javascript, la première accolade se met directement après l’instruction.

L’indentation se compose de 3 espaces (donc pas de tabulation).